

Neste relatório, apresentamos a implementação de um sistema de compartilhamento de arquivos peer-to-peer (P2P) usando sockets em Python. O sistema permite que vários peers se conectem a um servidor central e compartilhem informações sobre seus arquivos. Outros peers podem pesquisar e baixar esses arquivos diretamente dos peers que os possuem.

A arquitetura do sistema é composta por dois componentes principais: o servidor e os peers. O servidor atua como uma entidade centralizada que gerencia informações sobre os peers e seus arquivos. Os peers são os usuários do sistema que compartilham e baixam arquivos entre si.

Funcionalidades Implementadas:

O sistema implementa as seguintes funcionalidades principais:

1. JOIN: Um peer pode se juntar ao sistema enviando uma solicitação de JOIN para o servidor. Essa solicitação contém informações sobre o peer, como seu endereço IP, porta, pasta local onde os arquivos estão armazenados e a lista de arquivos que ele deseja compartilhar. O servidor armazena essas informações em uma estrutura de dados para referência futura.
2. SEARCH: Um peer pode pesquisar arquivos no sistema enviando uma solicitação de SEARCH para o servidor. A solicitação contém o nome do arquivo desejado. O servidor verifica sua estrutura de dados para identificar os peers que possuem o arquivo e retorna uma lista de endereços IP e portas desses peers.

Funcionalidades Não-Implementada:

3. DOWNLOAD: Um peer pode baixar um arquivo de outro peer. Para isso, ele fornece o endereço IP do peer e o nome do arquivo desejado. O peer que possui o arquivo recebe a solicitação de DOWNLOAD e envia o arquivo diretamente para o peer solicitante.

Implementação

A implementação do sistema foi realizada em Python usando sockets para comunicação entre o servidor e os peers. O código foi dividido em dois arquivos: `peer.py` e `server.py`.

O arquivo `peer.py` contém a lógica para interação dos peers com o servidor. Ele permite que os peers se juntem ao sistema, pesquisem arquivos disponíveis e realizem o download de arquivos de outros peers. O peer também mantém informações sobre seu próprio endereço IP, porta, pasta local e lista de arquivos compartilhados.

O arquivo `server.py` implementa a lógica do servidor. Ele recebe as solicitações dos peers, como JOIN, SEARCH e DOWNLOAD, e responde de acordo. O servidor armazena as informações dos peers em uma estrutura de dados e realiza a busca por arquivos no sistema.

O sistema permite que os peers se juntem, compartilhem informações sobre seus arquivos e realizem buscas e downloads entre si. Essa abordagem descentralizada de compartilhamento de arquivos oferece escalabilidade e eficiência, permitindo que os peers troquem arquivos diretamente, eliminando a necessidade de um servidor centralizado.

A implementação do sistema pode ser estendida para adicionar recursos adicionais, como upload de arquivos, autenticação de usuários e gerenciamento de conexões ponto a ponto

mais robusto. O código fornecido serve como um ponto de partida para explorar ainda mais o mundo do compartilhamento de arquivos peer-to-peer.